Contexto del negocio:

La tienda de ropa Boutique Talú, requiero para el mejoramiento de la atención a sus clientes una aplicación para gestionar su inventario, ventas, usuarios y clientes. La aplicación debe permitir a los usuarios autenticarse con diferentes roles (administrador, vendedor) y realizar operaciones CRUD en los productos del inventario, los usuarios y los cliente, así como realizar transacciones de venta.

Propósito de la aplicación:

El propósito de la aplicación es proporcionar una interfaz fácil de usar para que los empleados de la tienda gestionen el inventario y realicen ventas de ropa de manera eficiente. Además, la aplicación debe garantizar la seguridad de los datos mediante la autenticación de usuarios y la asignación de roles.

Solución:

Patrones de diseño a aplicar:

Creacionales:

Patrón Builder:

Clase de Implementación: Utilizaremos el patrón Builder para la creación de objetos complejos de productos de ropa, permitiendo una construcción paso a paso y la configuración de múltiples atributos, como tallas, colores y estilos.

Patrón Factory Method:

Clase de Implementación: Implementaremos el patrón Factory Method para crear instancias de objetos de tipo usuario según el rol especificado durante el proceso de autenticación. Esto nos permitirá crear objetos de diferentes subclases de usuarios (administrador, vendedor) sin exponer la lógica de creación directamente a la interfaz.

Patrón Singleton:

Clase de Implementación: Aplicaremos el patrón Singleton para garantizar que solo exista una instancia de la clase de conexión a la base de datos en toda la aplicación. Esto nos permitirá centralizar el acceso a la base de datos y evitar la creación de múltiples instancias que podrían causar problemas de concurrencia.

Estructurales:

Patrón Proxy:

Clase de Implementación: Emplearemos el patrón Proxy para controlar el acceso a los objetos de productos de alta demanda, como los de temporada. Esto permitirá gestionar su disponibilidad de manera eficiente y controlar el acceso a ellos para evitar exceso de tráfico.

Patrón Bridge:

Clase de Implementación: Implementaremos el patrón Bridge para separar la abstracción de los productos de ropa de su implementación. Esto facilitará la gestión de diferentes tipos de productos y su presentación en la interfaz de usuario, permitiendo una mayor flexibilidad en la estructura de la tienda.

Patrón Decorator:

Clase de Implementación: Utilizaremos el patrón Decorator para agregar funcionalidades adicionales a los productos del inventario de ropa, como descuentos o promociones. Esto nos permitirá extender dinámicamente la funcionalidad de los productos sin modificar su estructura base.

De Comportamiento:

Patrón Strategy:

Clase de Implementación: Aplicaremos el patrón Strategy para definir algoritmos específicos de cálculo de precios y descuentos para diferentes tipos de productos de ropa. Esto permitirá adaptar el cálculo del precio final según la estrategia de precios de la tienda, como descuentos por temporada o promociones especiales.

Patrón Observer:

Clase de Implementación: Utilizaremos el patrón Observer para notificar a los diferentes componentes de la aplicación sobre los cambios en el inventario de productos de ropa. Esto permitirá una actualización en tiempo real de la interfaz de usuario cuando se produzcan cambios en el inventario.

Patrón Command:

Clase de Implementación: Aplicaremos el patrón Command para encapsular las solicitudes del usuario como objetos, lo que permitirá parametrizar las operaciones y ejecutarlas de manera diferida o encolada. Por ejemplo, podríamos tener comandos para agregar productos al carrito de compra, eliminar productos, etc.

Operaciones a implementar:

Operaciones CRUD:

Crear, leer, actualizar y eliminar productos de ropa del inventario.

Crear, leer, actualizar y eliminar usuarios con diferentes roles (administrador, vendedor).

Operaciones transaccionales:

Realizar ventas, que implican actualizar el inventario y generar un recibo para el cliente.

Implementar ventas con descuentos para ciertos productos o en ciertos períodos de tiempo.

Permitir ventas a crédito, donde se establezca un plazo de pago para el cliente y se registren los pagos parciales.

Gestión de usuarios:

Los usuarios podrán registrarse en la aplicación especificando su nombre de usuario y contraseña.

El sistema permitirá a los usuarios autenticarse con sus credenciales.

Habrá dos roles de usuario: administrador y vendedor.

Los administradores podrán gestionar (crear, leer, actualizar, eliminar) usuarios y productos.

Los vendedores solo podrán realizar ventas y ver el inventario, sin poder modificar usuarios ni productos.